

## An efficient time algorithm for makespan objectives

Yücel Öztürkoglu

Department of International Logistics Management, Yaşar University, Turkey  
Email: yucel.ozturkoglu@yasar.edu.tr

(Received May 18, 2015; in final form June 25, 2015)

---

**Abstract.** This paper focuses on a single machine scheduling subject to machine deterioration with rate-modifying activities (RMA). The motivation for this study stems from the automatic-production line problem with one machine. The main question is to find the sequence in which jobs should be scheduled, how many maintenance activity (RMA) to use, if any, and where to insert them in the schedule during the time interval with optimal makespan objective. This problem is known to be NP-hard and we give concise analyses of the problem and provide polynomial time algorithms to solve the makespan problem. We also propose an algorithm which can be applied to some scheduling problems with the actual processing time of job nonlinearly based on its position.

**Keywords:** Scheduling; deteriorating jobs; rate-modifying-activity; single machine; makespan.

**AMS Classification:** 90B35,68M20

---

### 1. Introduction

In this paper, we study scheduling jobs and preventive maintenance under two new phenomena in the scheduling literature. First one is a deteriorating job. Deteriorating jobs are tasks which need more time and more efforts to complete the process than when they are done earlier. Browne-Yechiali [1] and Mosheiov [2] were first introduced deteriorating jobs in the scheduling area. After a while, deteriorating jobs were getting more attention. Some milestone studies in the literature are respectively; Kubiak and Vende [3] investigated the computational complexity of makespan under deterioration. They developed a heuristic and branch-and-bound algorithm for the problem. Kovalyov and Kubiak [4] presented a fully polynomial approximation scheme for a single machine scheduling problem to minimize makespan of deteriorating jobs. Cheng and Ding [5] studied a single machine problem to minimize makespan with deadlines and increasing rates of processing times. All these researchers consider

simple linear deterioration of processing times;

$$p_i(t) = a_i t \quad (1)$$

In this study, we use different deterioration rate ( $\alpha$ ) which changes the processing time of the jobs nonlinearly based on its position.

Deterioration of the jobs is commonly due to machine wear. To prevent the wear, a machine needs a preventive activity that changes the production rate of the equipment. Qi et al [6] considered a problem where multiple maintenance activities need to be scheduled with jobs on a single machine. Lee and Leon [7] called this activity as a rate-modifying activity (RMA). RMA is a maintenance task performed before a machine fails. After giving RMA, the performance of the machine is assumed to be normal. Lee and Lin [8], He et al. [9], Mosheiov and Sidney [10], Gordon and Tarasevich [11], Wang and Wang [12] studied RMA with different perspectives.

Combining RMA and deteriorated jobs are new research area for scheduling problems. Firstly, Lodree and Geiger [13] integrated time dependent processing times and RMA for assigning a single RMA to a position. They showed that a single RMA should be inserted in the middle of the optimal job sequence to minimize makespan. Lately, Ozturkoglu and Bulfin [14] proposed a mathematical model for problem

$$1 | p_{ij} = (1 + \alpha_j)^{i-1} p_j, rm | C_{\max} \quad (2)$$

They showed that as the number of jobs increases, the computational time to solve the problem increases dramatically with the mathematical model. So their model is useless for large size problems. One year later, Ozturkoglu and Bulfin [15] extended their model and defined RMA as workers break time. They considered physiological condition of workers and added physiological constraints to their model with makespan objectives. But the NP-hardness of the considered problem is still open.

In this paper, we extended Ozturkoglu and Bulfin [14]'s study and we prove that the same problem can be solved by polynomial time algorithm. We propose polynomial time algorithms for makespan objectives. Also, we proofed several theorems to support our results.

## 2. Problem statement

The problem under consideration is motivated by the problem of automatic-production line problem with one machine. The production process is stopped during the rate-modifying activity, leading to an increase of the completion times of the following jobs. The scheduler must decide on whether or when to schedule the rate-modify activity during the scheduling horizon to optimal performance measures.

The problem we study in this paper is to schedule a set of  $n$  independent jobs  $J = \{J_1, J_2, \dots, J_n\}$  and should assign at least one RMAs for a single machine. All jobs are available for processing at all times. The machine can do only one job at a time. Each job has a deterioration rate  $\alpha$  which reflects a delay time in processing jobs. We assume that the deterioration rate  $\alpha$  has the same effect on processing times of different jobs and it changes the processing time of the job *nonlinearly based on its position*. The main problem is to find the sequence in which jobs should be scheduled, how many RMAs (maintenance activity) to use, if any, and where

to insert them in the schedule. We use Ozturkoglu and Bulfin's [14] parameters in our algorithms.

### Model Parameters:

$n$  is the number of jobs to be sequenced,  $i$  indicates the position number which is from 1 to  $n$ ,  $k$  indicates the position number which is from 0 to  $n$  ( $k=0$  is initial position),  $j$  indicates the job number which is from 1 to  $n$ ,  $\alpha$  ( $0 < \alpha \leq 1$ ), is the constant deterioration rate of jobs when delayed by one position,  $q$  is the fixed period of time to perform an RMA,  $p_j$  is the initial processing time of job  $j$  before deterioration,  $p_{ji}$  is the processing time of job  $j$  if done  $i$  positions after an RMA or the initial position, and the formulation of  $p_{ji}$  is;

$$p_{ji} = (1 + \alpha)^{i-1} p_j \quad (3)$$

Using three field notations of Graham et al. [16], the problem can be denoted a.

$$1 | p_{ij} = (1 + \alpha_j)^{i-1} p_j, rm | C_{\max} \quad (4)$$

## 3. Polynomial time algorithms for makespan

In this section, we develop some fundamental properties with special case for the unit processing time problem. And, then we develop polynomial time algorithms to minimize makespan when the basic processing time of all jobs is identical. The following theorems are particularly useful in this paper.

**Theorem 1.** *Balanced schedules are optimal under  $r = 0$ , if  $k_i = id$ ,  $i = 1, \dots, m$ .*

$$C_{\max} = \sum_{i=1}^{k_1} p(1 + \alpha)^{i-1} + \sum_{i=1}^{(k_2 - k_1)} p(1 + \alpha)^{i-1} + \dots + \sum_{i=1}^{(n - (k_1 + k_2 + k_3))} p(1 + \alpha)^{i-1} + mq \quad (5)$$

Since  $mq$  is constant, we need to choose  $k_1, k_2, \dots, k_m$  to minimize  $C_{\max}$ . This happens when  $k_1, k_2 - k_1, \dots, n - k_m$  are as close to equal as possible. Let  $n = d(m + 1) + r$  and  $d$  is the integer.

**Proof.** Let  $n$  is the number of jobs,  $k$  is the number of given RMAs and  $q$  is the length of

RMA. The other two important parameters are  $r$  which is the remainder of  $\frac{n}{k+1}$  and  $d$  which is the division of  $\frac{n}{k+1}$ .

Assume  $S$  is an optimal schedule with  $k_j = d+1$  and  $k_l = d-1$  and  $S'$  the same schedule except  $k_j = d$  and  $k_l = d$ . All terms of  $C_{\max}$  for both  $S$  and  $S'$  are equal except those containing  $k_j$  and  $k_l$  for;

$$\begin{aligned} S &\Rightarrow \sum_{i=1}^{k_j} p(1+\alpha)^{i-1} + \sum_{i=1}^{k_l} p(1+\alpha)^{i-1} + q \\ &= p + p(1+\alpha)^1 + \dots + p(1+\alpha)^{d-1} + p(1+\alpha)^d + q + p \\ &\quad + p(1+\alpha)^1 + \dots + p(1+\alpha)^{d-2} \end{aligned} \quad (6)$$

$$\begin{aligned} S' &\Rightarrow \sum_{i=1}^{k_j} p(1+\alpha)^{i-1} + \sum_{i=1}^{k_l} p(1+\alpha)^{i-1} + q \\ &= p + p(1+\alpha)^1 + \dots + p(1+\alpha)^{d-1} + q \\ &\quad + p + p(1+\alpha)^1 + \dots + p(1+\alpha)^{d-1} \end{aligned} \quad (7)$$

$$S - S' = p(1+\alpha)^d - p(1+\alpha)^{d-1} \quad (8)$$

$C_{\max}(S) - C_{\max}(S') \geq 0$ . This means that,  $S'$  is better than  $S$  and contradicting  $S$  is optimal.  $\square$

If  $S$  has more than two  $k_i$  different than  $d$ , or has a  $k_i > d+1$  or  $k_i < d-1$ , we can use the same arguments to show it is not as good as  $S'$ . As schedule with  $k_i = id$ ,  $i = 1, \dots, m$  is called exactly balanced. If  $r \neq 0$  then it is not possible to have an exactly balanced schedule.

**Definition 1.** We will call a schedule with  $k_i$  is a balanced schedule;

$$k_i = \begin{cases} id & i = 1, \dots, m+1-r \\ id + [i - (m+1-r)] & i = r+1, \dots, m \end{cases}$$

Actually, any schedule with  $r$  groups having  $d+1$  jobs and  $m+1-r$  groups with  $d$  jobs is balanced and all have the same makespan.

**Theorem 2.** For  $1/p_{ij} = (1+\alpha)^{i-1} p_j / C_{\max}$ , LPT (largest processing time) sequence minimizes the makespan.

**Proof.** Since  $p_{nj} = (1+\alpha)^{n-1} p_j$  and suppose schedule  $S$  minimizes makespan and is not in LPT order, then there must be a pair of jobs in  $S$ , say job  $i$  and job  $j$ , with job  $i$  immediately after job  $j$  in the  $k^{th}$  and  $k+1^{st}$  positions, and  $p_i < p_j$ , where  $i \neq j$ ,  $0 < i \leq n$  and  $0 < j \leq n$ . Let  $B$  be the set of jobs before job  $i$  and  $j$ , and  $A$  the set of jobs after job  $i$  and  $j$ . Let  $C(A)$  and  $C(B)$  be the sum, of processing times in sets  $A$  and  $B$  respectively. Now consider the schedule  $S'$ , where  $S'$  the same as  $S$  except job  $i$  and  $j$  have been interchanged and the sets of jobs  $A$  and  $B$  are in the same position in both schedules.  $S = (\pi, i, j, \pi')$  and  $S' = (\pi, j, i, \pi')$  where  $\pi$  and  $\pi'$  denote partial sequences.

The makespan for  $S$  is;

$$C(S) = C(B) + p_{(n-1)i} + p_{nj} + C(A) \quad (9)$$

$$C(S) = C(B) + (1+\alpha)^{n-2} p_i + (1+\alpha)^{n-1} p_j + C(A) \quad (10)$$

and the makespan for  $S'$  is;

$$C(S') = C(B) + p_{(n-1)j} + p_{ni} + C(A) \quad (11)$$

$$C(S') = C(B) + (1+\alpha)^{n-2} p_j + (1+\alpha)^{n-1} p_i + C(A) \quad (12)$$

Subtracting equation  $C(S)$  from  $C(S')$  we get  $C(S) - C(S') = \alpha(1+\alpha)^{n-2} (p_j - p_i) > 0$  (13)

This implies the makespan of  $S'$  is smaller than  $S$ , which contradicts the assumption that  $S$  was optimal. Therefore, an optimal solution must be in LPT.  $\square$

**Theorem 3.** Between given two RMA, the sequence of the jobs is always LPT order.

**Proof.** Let  $C(B)$  be the total completion time of the jobs before given RMA and  $C(A)$  be the total completion time of the jobs after given second RMA.

Now consider the schedule  $S = \{\pi, rma, i, j, k, rma, \pi'\}$  (SPT order) and

$S' = (\pi, rma, k, j, i, rma, \pi')$  (LPT order) where  $\pi$  and  $\pi'$  denote partial sequences.

The sets of jobs  $A$  and  $B$  are in the same position in both schedules. We assume that schedule  $S$  is an optimal schedule.

$$S' = (\pi, rma, i, j, k, rma, \pi')$$

Let's assume that,  $p_i < p_j < p_k$  and after RMA first job assign  $(n-1)^{th}$  position. The  $m$  is defined as a duration of maintenance activity (RMA).

The makespan for  $S$  is;

$$C_{\max}(S_{n-1}) = C(B) + m + p_i \tag{14}$$

$$C_{\max}(S_n) = C(B) + m + p_i + p_j(1 + \alpha)^{n-1} \tag{15}$$

$$C_{\max}(S_{n+1}) = C(B) + m + p_i + p_j(1 + \alpha)^{n-1} + p_k(1 + \alpha)^n \tag{16}$$

$$C_{\max}(S_{n+2}) = C(B) + m + p_i + p_j(1 + \alpha)^{n-1} + p_k(1 + \alpha)^n + m + C(A) \tag{17}$$

and the makespan for  $S'$  is;

$$C_{\max}(S'_{n-1}) = C(B) + m + p_k \tag{18}$$

$$C_{\max}(S'_n) = C(B) + m + p_k + p_j(1 + \alpha)^{n-1} \tag{19}$$

$$C_{\max}(S'_{n+1}) = C(B) + m + p_k + p_j(1 + \alpha)^{n-1} + p_i(1 + \alpha)^n \tag{20}$$

$$C_{\max}(S'_{n+2}) = C(B) + m + p_k + p_j(1 + \alpha)^{n-1} + p_i(1 + \alpha)^n + m + C(A) \tag{21}$$

Subtracting  $C(S)$  from  $C(S')$  we get;

$$C_{\max}(S) - C_{\max}(S') = \left| (1 + \alpha)^n - 1 \right| (p_k - p_i) > 0 \tag{22}$$

The schedule  $S'$  is better than schedule  $S$  which contradicts the assumption that  $S$  was optimal. This means that between given two RMA, the

schedule is always LPT order, like schedule  $S' = \{\pi, k, j, i, \pi'\}$ .  $\square$

**Theorem 4.** The algorithm finds an optimal solution for the problem  $1/ p_{ij} = (1 + \alpha)^{i-1} p, rm / C_{\max}$  with time complexity  $O(n \log n)$ .

**Proof.** The algorithm is based on the division of the  $\frac{n}{k+1}$ . Before we construct the algorithm, we present two equations which are based on remainder of the  $\frac{n}{k+1}$ .

$$C_{\max} = \begin{cases} \text{If } r = 0, & (k+1)p \left[ 1 + \underbrace{\sum_{i=1}^{d-1} (1 + \alpha)^i}_A \right] + kq \\ \text{If } r \neq 0, & (k+1)p \left[ 1 + \underbrace{\sum_{i=1}^{d-1} (1 + \alpha)^i}_A \right] + kq + 2p(1 + \alpha)^d \end{cases}$$

To simplify the general equation, we use notation  $A$  to calculate the term;

$$p \left[ 1 + \sum_{i=1}^{d-1} (1 + \alpha)^i \right] \tag{23}$$

The general equation is:

$$C_{\max} = \begin{cases} r = 0 & (k+1)A + kq \\ r \neq 0 & (k+1)A + kq + 2p(1 + \alpha)^d \end{cases} \tag{24}$$

The procedure for the algorithm is stated as follows;

**Algorithm**

Step1. Determine  $n$ ,  $\alpha$  and  $q$ .

Step2. Calculate remainder (r) of  $\frac{n}{k+1}$ .

Step3. Calculate division (d) of  $\frac{n}{k+1}$ .

Step4. Determine  $A$  based on given equation;

$$A = p \left[ 1 + \sum_{i=1}^{d-1} (1 + \alpha)^i \right].$$

Step5. Based on remainder (r), calculate  $C_{\max}^i$

$$\begin{aligned} r = 0 & \quad (k+1)A + kq \\ r \neq 0 & \quad (k+1)A + kq + 2p(1 + \alpha)^d \end{aligned}$$

Step6. For each possible RMA assignment,  $k = 2, \dots, (n - 1)$  do Step 2, 3, 4, 5 and 6.

Step7. For each possible RMA, take smallest  $C_{\max}$ . This  $C_{\max}$  is the optimal solution.

$$C_{\max} = \min \{C_{\max}^1, C_{\max}^2, \dots, C_{\max}^{n-1}\}$$

The algorithm easily finds the optimal  $C_{\max}$  in a reasonable time for large numbers of problem. Besides optimal solution, the algorithm tells how many RMA needed and where should we put them.  $\square$

Furthermore the algorithm, if manager wants to give an exact number of maintenance activities, then we will use another equation which is also given an optimal solution of the makespan.

**The general equation for exact (m) number of RMAs.**

**Case 1.** If m is an even number then the general equation will be:

$$\begin{aligned} \Rightarrow & 2 \sum_{i=1}^d p(1+\alpha)^{i-1} + 2^{\left\lceil \frac{(n-2d)}{(k-2)+1} \right\rceil} p(1+\alpha)^{i-1} \\ & + 2^{\left\lceil \frac{(n-4d)}{(k-4)+1} \right\rceil} p(1+\alpha)^{i-1} + 2^{\left\lceil \frac{(n-6d)}{(k-6)+1} \right\rceil} p(1+\alpha)^{i-1} \\ & + \dots + 2^{\left\lceil \frac{(n-md)}{(k-m)+1} \right\rceil} p(1+\alpha)^{i-1} + mq \end{aligned} \tag{25}$$

**Case 2.** If m is an odd number then the general equation will be:

$$\begin{aligned} \Rightarrow & 2 \sum_{i=1}^d p(1+\alpha)^{i-1} + 2^{\left\lceil \frac{(n-2d)}{(k-2)+1} \right\rceil} p(1+\alpha)^{i-1} \\ & + 2^{\left\lceil \frac{(n-4d)}{(k-4)+1} \right\rceil} p(1+\alpha)^{i-1} + 2^{\left\lceil \frac{(n-6d)}{(k-6)+1} \right\rceil} p(1+\alpha)^{i-1} \\ & + \dots + 2^{\left\lceil \frac{(n-(m-1)d)}{(k-(m-1))+1} \right\rceil} p(1+\alpha)^{i-1} + mq \end{aligned} \tag{26}$$

**4. Conclusion**

In this paper, the main question is to find the sequence in which jobs should be scheduled, how

many RMAs (maintenance activity) to use, if any, and where to insert them in the schedule. We give a concise analysis of the problem which is first proposed by Ozturkoglu and Bulfin [14] and provide simpler algorithms for the problem. Our main findings for the makespan minimization problem provide polynomial time algorithms to solve the very large problem's optimally with time complexity  $O(n \log n)$ . Also we develop some fundamental properties and polynomial algorithm for the unit processing time problem. Moreover, this study has certain industrial applications such as maintenance planning and production scheduling. The maintenance engineers and production engineers are the likely users. The proposed algorithm can be easily applied to any scheduling problem if there is a maintenance activity is available.

**References**

- [1] Browne, S., Yechiali, U., Scheduling deteriorating jobs on a single processor, Operations Research Society, 38, 495-498 (1990).
- [2] Mosheiov, G., V-shaped policies for scheduling deteriorating jobs, Operations Research, 39, 979-991 (1991).
- [3] Kubiak, W., Van de Velde, S., Scheduling deteriorating jobs to minimize makespan, Naval Research Logistics, 45, 511-523 (1998).
- [4] Kovalyov, Y. M., Kubiak, W., A fully polynomial approximation scheme for minimizing makespan of deteriorating jobs, Journal of Heuristics, 3, 287-297 (1998).
- [5] Cheng, T. C. E., Ding, Q., Single machine scheduling with deadlines and increasing rates of processing times, Acta Informatica, 36, 673-692 (2000).
- [6] Qi, X., Chen, T., Tu, F., Scheduling with the maintenance on a single machine, Working Paper, Department of Computer System Sciences, Nankai University, China (1997).
- [7] Lee, C. Y., Leon, V. J., Machine scheduling with a rate-modifying activity, European Journal of Operational Research, 128, 119-128 (2001).
- [8] Lee, C. Y., Lin, C. S., Single-machine scheduling with maintenance and repair rate-modifying activities, European Journal of Operations Research, 135, 493-513 (2001).

- [9] He, Y., Ji, M., Cheng, T. C. E., Scheduling with a restricted rate-modifying activity, *Naval Research Logistics*, 52:361-369 (2005).
- [10] Mosheiov, G., Sidney, J. B., Scheduling a deteriorating maintenance activity on a single machine, *Journal of the Operational Research Society*, 61, 882-887 (2010).
- [11] Gordon, V. S., Tarasevich, A. A., A note: common due date assignment for a single machine scheduling with the rate-modifying activity, *Computer Operations Research*, 36, 325-328 (2009).
- [12] Wang, X. Y., Wang, M. Z., Single machine common flow allowance scheduling with a rate-modifying activity, *Computers and Industrial Engineering*, 59, 898-902 (2010).
- [13] Lodree, E. J., Geiger, C. D., A note on the optimal sequence position for a rate-modifying activity under simple linear deterioration, *European Journal of Operational Research*, 201, 644-648 (2010).
- [14] Ozturkoglu, Y., Bulfin, R. L., A unique integer mathematical model for scheduling deteriorating jobs with rate-modifying-activities on a single machine, *International Journal of Advanced Manufacturing Technology*, 57, 753-762 (2011).
- [15] Ozturkoglu, Y., Bulfin, R. L., Scheduling jobs to consider physiological factors, *Human Factors and Ergonomics in Manufacturing & Service Industries*, 22(2), 113-120 (2012).
- [16] Graham, R. L., Lawler, E. L., Lenstra, J. K., Rinnooy, K. A. H. G., Optimization and approximation in deterministic sequencing and scheduling: A Survey, *Annals of Discrete Mathematics*, 5, 287-326 (1979).

*Yücel Öztürkoğlu has been an Assistant Professor in the Department of International Logistics Management in Yasar University. She has a BA in Industrial Engineering (Çankaya University/Turkey), M.Sc. in MBA (Erciyes University/ Turkey), MIS and a PhD degree in Industrial Engineering (Auburn University/ USA). Her research interest includes scheduling, decision making models, supply chain management and production management.*